

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently Amended) A process for transferring pen data between unmanaged and managed code on a computing device, the unmanaged code being code native to and executed directly by a processor of the computing device, the managed code being executed in a common language run-time environment of a framework operating on the computing device, the common language run-time environment of the framework executing the managed code independent of a type of the processor of the computing device, the process comprising the steps of:

receiving pen data in a component on the computing device written in unmanaged code, the pen data being generated by a digitizer of the computing device upon movement of a stylus with respect to a surface of the digitizer, the pen data including at least one location on the digitizer of the stylus;

transferring information related to said pen data to a shared memory on the computing device designated to be shared between unmanaged code and managed code;

transferring, by the component written in unmanaged code, a pointer that points to said information in said shared memory to an application on the computing device written at least in part in managed code; and

retrieving, by the application written at least in part in managed code, said information from said shared memory by way of the transferred pointer.

2. (Original) The process according to claim 1, further comprising the steps of: transferring additional information from said at least in part managed application to said shared memory;

transferring a pointer that points to said additional information to said component;

retrieving said additional information from said shared memory.

3. (Original) The process according to claim 1, further comprising the step of:

using a P-invoke style interface crossing between unmanaged code and managed code.

4. (Original) The process according to claim 1, further comprising the step of: exchanging information through a COM interface.

5. (Original) The process according to claim 1, said component being a pen services component.

6. (Original) The process according to claim 1, said application including a pen input managed client.

7. (Original) The process according to claim 1, said component receiving input from at least one pen device driver.

8. (Currently Amended) A system for transferring information between unmanaged code and managed code on a computing device, the unmanaged code being code native to and executed directly by a processor of the computing device, the managed code being executed in a common language run-time environment of a framework operating on the computing device, the common language run-time environment of the framework executing the managed code independent of a type of the processor of the computing device, the system comprising:

a shared memory on the computing device designated to be shared between unmanaged code and managed code;

a component on the computing device that is written in unmanaged code and that receives pen data, the pen data being generated by a digitizer of the computing device upon movement of a stylus with respect to a surface of the digitizer, the pen data including at least one location on the digitizer of the stylus, the component transferring and transfers information relating to said pen data to said shared memory and transfers transferring a pointer that points to said

information in the shared memory to an application on the computing device having managed code;

said application having managed code receives receiving said pointer and ~~obtains~~ obtaining said information from said shared memory by way of the transferred pointer.

9. (Original) The system according to claim 8, said component exposing a COM interface.

10. (Original) The system according to claim 8, said application using a P-Invoke-style command.

11. (Original) The system according to claim 8, said component including a pen services component.

12. (Original) The system according to claim 8, further comprising:
at least one pen device driver sending information to said component.

13. (Original) The system according to claim 8, further comprising:
said application including a pen input managed client.

14. (Currently Amended) A computer-readable storage medium having a program stored thereon for transferring information related to ink between an unmanaged component and an application including managed code on a computing device, the unmanaged component being native to and executed directly by a processor of the computing device, the managed code of the application being executed in a common language run-time environment of a framework operating on the computing device, the common language run-time environment of the framework executing the managed code independent of a type of the processor of the computing device, said program comprising the steps of:

receiving pen data in ~~[[a]] the unmanaged component written in unmanaged code on the~~ computing device, the pen data being generated by a digitizer of the computing device upon movement of a stylus with respect to a surface of the digitizer, the pen data including at least one location on the digitizer of the stylus;

transferring information related to said pen data to a shared memory on the computing device designated to be shared between unmanaged code and managed code;

transferring, by the unmanaged component, a pointer that points to said information in said shared memory to [[an]] the application written at least in part in managed code; and

retrieving, by the managed code of the application, said information from said shared memory by way of the transferred pointer.

15. (Currently Amended) The computer-readable storage medium according to claim 14, said program further comprising the steps of:

transferring additional information from said at least in part managed application to said shared memory;

transferring a pointer that points to said additional information to said component;

retrieving said additional information from said shared memory.

16. (Currently Amended) The computer-readable storage medium according to claim 14, said program further comprising the step of:

using a P-invoke style interface crossing between unmanaged code and managed code.

17. (Currently Amended) The computer-readable storage medium according to claim 14, said program further comprising the step of:

exchanging information through a COM interface.

18. (Currently Amended) The computer-readable storage medium according to claim 14, said component being a pen services component.

19. (Currently Amended) The computer-readable storage medium according to claim 14, said application including a pen input managed client.

20. (Currently Amended) The computer-readable storage medium according to claim 14, said component receiving input from at least one pen device driver.